Developing Adjustable Walking Patterns for Natural Walking in Humanoid Robots

James M. Jeanne, Princeton University

Mentor: Dr. Adrian Stoica, Jet Propulsion Laboratory

19 August 2004

Because of the potential for interaction with humans, research in humanoid robotics has made significant progress in recent years. It has long been imagined that some day, humanoid robots will serve people with disabilities, improve search-and-rescue efforts in dangerous conditions, and perhaps even replace humans in doing our everyday tasks. The ability of a humanoid robot to do these tasks, however, relies on its locomotion capabilities. On a large scale, the robot must be able to walk in many types of terrain if it is to operate successfully in diverse conditions. A walking pattern was thus developed that allows for the stride length, the single-support balance point, turning angle, and the maximum height of the swing foot to be modified dynamically. This will then enable the robot to undergo a learning process to determine how certain parameters affect its walking and will enable higher-level control systems that use inputs from vision and tactile sensors to make its walking robust to many different environments.

Introduction

One of the most exciting aspects of humanoid robotics research is the potential for a wide range of applications. With their shape and movement inspired by that of humans, they are poised to operate successfully in a human environment, and possibly to perform any task that a human can perform. However, before a robot can do this, it must have a robust means of locomotion. Several walking-pattern generation methods have been studied, including Zero Moment Point (ZMP) walking, where the center of gravity is maintained over the robot's support structure at all times;[1] rhythmic walking, where the walking frequency is continuously adjusted to maintain the robot's balance;[2] and ballistic walking, where the motors do not power the legs when they swing forward.[3] All of these methods have proven successful for generating stable walking in either simulation or demonstration.

Furthermore, several techniques have been suggested for increasing the robustness of humanoid walking patterns. These methods include unsupervised learning, where a careful search of parameter values is performed to find the best value for the current conditions,[3] as well as fostering and reinforcement learning, where parameter values are determined through feedback from a human trainer or a predefined learning function.[4] Reinforcement learning has been shown to be especially effective in developing stable walking patterns in humanoid robots.[2]

In this paper, I describe the process of developing a stable ZMP walking pattern for a Fujitsu HOAP-2 (Humanoid for Open Architecture Platform) Humanoid Robot. The HOAP-2 robot has 21 degrees of freedom: 6 degrees of freedom in each leg, 4 in each arm, and one in its waist (See figure 1). In

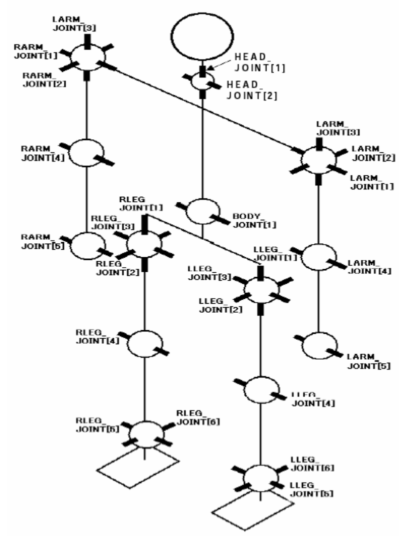

Figure 1: Schematic diagram of the HOAP-2 robot. Each degree of freedom is shown as a black line through a sphere.[8]

addition, there are 4 pressure sensors in each foot, and an accelerometer and gyroscope inside the torso. The robot is controlled using a command PC (2.4GHz Pentium 4) with a USB communications link to the robot. I used the ZMP method of walking because of its ease of development and speed of implementation. I designed the walking pattern with a set of parameters that define the size of each step, the height of each step, the angle the robot turns per step, and the position of the feet when the robot is standing still. Each parameter was made to be adjustable with the intention of using a reinforcement learning process to determine its optimal value for different environmental conditions.

Because of the dependence of ZMP walking on an accurate model of the kinematics of the robot, it was necessary to develop a mathematical model of the motion of the robot before designing the walking pattern and testing it on the HOAP-2. The modeling process is described in the Findings section, the results of testing this pattern on the actual robot are described in the Discussion section, and the details of the mathematical model of the HOAP-2's kinematics are described in the Methods section.

Findings

Mathematical Model of the Legs of the HOAP-2

Before beginning work on the development of a basic walking pattern for the HOAP-2 humanoid robot, I developed a mathematical model of the kinematics of its legs. This forward-kinematics model mapped all six of the angles of each degree of freedom in each leg to the position and orientation of each foot. The mapping is thus

$$\begin{bmatrix} \theta_{Hip,x} \\ \theta_{Hip,y} \\ \theta_{Hip,z} \\ \theta_{Knee,x} \\ \theta_{Ankle,x} \\ \theta_{Ankle,y} \end{bmatrix} \rightarrow \begin{bmatrix} Foot_x \\ Foot_y \\ Foot_z \\ Foot_\varphi \\ Foot_\theta \\ Foot_\psi \end{bmatrix},$$

where each θ represents one of the degrees of freedom in each leg, the x, y, and z parameters of Foot describe the three-dimensional position of the bottom of the foot directly below the ankle, and the φ, θ, and ψ parameters of Foot describe the three-dimensional orientation of the foot. Figure 2 gives the relationship of the angles in the vector above to the actual joints in the robot. In the mathematical model, I represented each degree of freedom with a three-dimensional rotation matrix and each joint with a simple translation vector. With this model, I could quickly and easily simulate the effects of basic joint adjustments without actually taking the time to run them on the robot. The exact kinematics model is given in the Methods section.

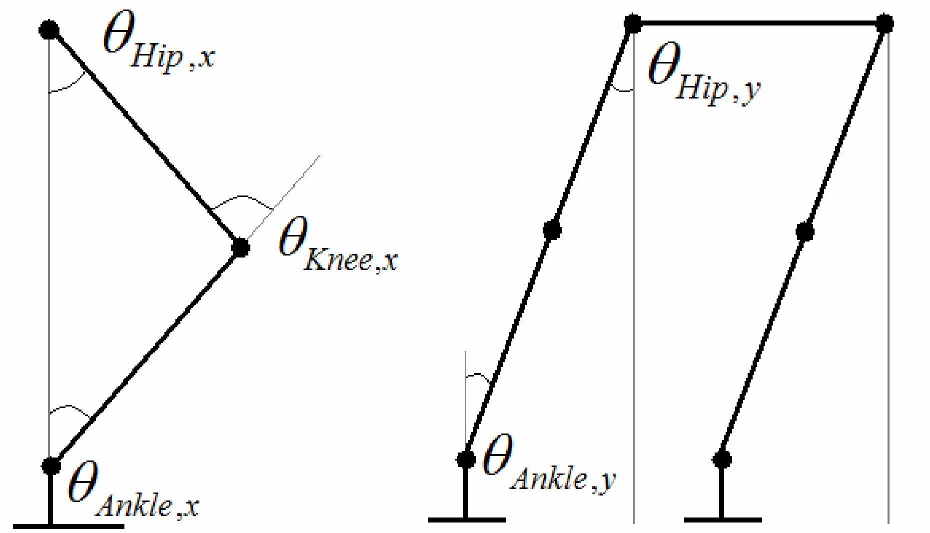

Figure 2: Relationship of angle definitions in the kinematics model to the actual joints of the HOAP-2. The left diagram represents the angles that produce motion in the sagittal plane (front-back) and the right diagram represents the angles that produce motion in the lateral plane (left-right). $\theta_{hip,z}$, the rotational motion of the hip, is not shown.

While the forward-kinematics model was helpful for early experimentation with motion, an even more useful model was the inverse-kinematics model. This model mapped the position and orientation of the feet to the angles of each degree of freedom of the leg, exactly the opposite of the forward-kinematics model. Because of the nonlinear combinations of trigonometric functions in the forward-kinematics model, it was not straightforward to find a closed-form solution for the inverse model.

Although there are several ways to approach this problem, some of them have serious problems that make them difficult to use on a platform with a sampling time of 1ms. The brute-force method of MATLAB's nonlinear equation solver ("fsolve"), proved to be too slow to implement on this real-time platform. Indeed, each calculation took approximately 18ms in the average case, precluding this method from use on the HOAP-2. To speed up individual calculation times, a neural network can be extensively trained to simulate the inverse relationship, but requires an unknown amount of training time.[5] I found that the fastest technique in terms of both development time and performance time is to simplify the problem by maintaining a constant orientation of the foot (i.e. setting $Foot_\varphi = 0$, $Foot_\theta = 0$, and $Foot_\psi = 0$), making a closed-form solution feasible.

By adopting this restriction to the system, only three of the degrees of freedom in each leg needed to be determined through the inverse-kinematics description; the remaining three could easily be determined from the others. I was thus able to reduce the mapping to

$$\begin{bmatrix} \theta_{Hip,x} \\ \theta_{Hip,y} \\ \theta_{Knee,y} \end{bmatrix} \rightarrow \begin{bmatrix} Foot_x \\ Foot_y \\ Foot_z \end{bmatrix},$$

which yielded a simple, closed-form solution quite readily. Furthermore, the calculation of each angle can be completed in well under 1ms, making it ideal for implementation on the real-time platform. The derivation of this closed-form solution is given in full in the Methods section.

Development of Walking Patterns

With a fast and accurate inverse-kinematics model of the humanoid robot's legs, I could then begin to develop a walking pattern that could be described by the position of the feet, rather than the angle of each joint. Based on the work of Huang, et. al.,[6] it is clear that smooth-transitioning trajectories are desirable in trajectory design. In addition, through my own experimentation, I found that sudden jerking motions in the leg can generate transient effects that can take many seconds to die out, thus significantly decreasing the stability of the robot. Consequently, I used third-order spline interpolation, a technique that defines a continuous and differentiable curve between an arbitrary number of points.[6]

Based on the well-researched concept of the Zero Moment Point (ZMP)—which is the point where all torque vectors cancel out to zero, also known as the center of gravity[7]—I defined a simple but stable walking trajectory. Since the idea of ZMP stability is to keep the center of gravity directly over the center of the support structure (in this case the robot's feet), the body of the robot needs to sway from one side to the other as it takes steps. To simplify the trajectory, I separated the walking pattern into two distinct stages. The first stage shifts the robot's center of gravity to be over the support leg and the second stage swings the robot's other leg forward. Smooth walking was thus obtained by executing this cycle on alternating feet repeatedly.

I defined the motion in each stage by defining several critical positions of the robot's feet (with respect to the center of the waist) and interpolating between them using the spline technique described above. Then, it was simply a matter of running the position values through the inverse-kinematics model to determine the appropriate joint angles for each degree of freedom.

In addition to the ease of design and speed of calculation, this method of describing a walking trajectory had another advantage. I could easily change the points that were interpolated to define the trajectory. This meant that the walking pattern was highly adjustable, because parameters such as stride length (how far the robot steps in one cycle), sway distance (how far the robot shifts its weight to one side when the other leg swings forward), step clearance (how high the robot lifts its feet when it steps), as well as the home position of its feet (the location of the feet beneath the robot when it stands still, to which all other positions are relative) could all be readily changed.

Development of Additional Motion Patterns

To further increase the motion capability of the HOAP-2, I used exactly the same procedures for its arms as I did for its legs. Although each arm has 4 degrees of freedom, the rotational joint of the shoulder needed to be fixed so that the elbow joint was able to bend towards the center of the body, enabling the hands to grasp large objects. Thus, only three degrees of freedom were mapped to the three dimensions of the position coordinates and the inverse kinematics could be determined in closed form in a manner similar to that of the legs. Four motions were developed: those for grasping an object, lifting the object, lowering the object, and releasing the object. These motions, combined with the walking motions, provided the robot with the ability to perform basic assembly tasks.

Discussion

Results

In testing the walking pattern on the HOAP-2, I have found that it is very stable (in that it does not fall over or trip itself up) as long as it remains on a smooth, horizontal surface. During the initial tests, there was a problem that caused the robot to wobble significantly and its feet to slip each time it put its right foot down. This turned out to be a result of the fact that there are about 1 to 2 degrees of play in each servo. As the right foot was lowered to the ground after swinging forward, the play in the lateral and sagittal hip joints meant that the left back corner of the foot touched before the rest of the bottom surface, and caused the robot to wobble significantly on its left foot, sometimes enough for the robot to fall over. I compensated for the error in the swing foot by increasing the distance from the hip to the bottom of the support foot by 0.5cm. This meant that the swing foot didn't touch the ground until it was supposed to.

Furthermore, the adjustability of the walking cycle proved to be successful as well. I could adjust the stride length to be anywhere from 0-6cm without causing the robot to wobble or to show signs of instability. When the arms were moved from the side of the robot to the front, the change in weight distribution caused the robot to fall forward when it started walking. This was easily fixed by shifting the home position of the robot's feet forward, thus bringing the center of gravity back to the center of the feet. With this change implemented, the robot walked naturally again.

Future Work

This ability to move the center of gravity as well as other parameters of the robot's walk is essential for the future development of a dynamically stabilized walking pattern. As was described in the previous paragraph, even lifting its arms up in front of the robot is enough to significantly change its center of gravity. Thus, any heavy lifting or carrying that the robot performs could potentially throw it off balance, and so a dynamic balance control system would be essential if the humanoid is to do any useful work. By polling the pressure sensors on the bottom of the feet, the Zero Moment Point can be calculated every millisecond, allowing for minute adjustments to the foot locations to correct for changes in the robot's center of gravity.

Other useful high-level control systems could be implemented as well, including a stride length controller that uses visual inputs to determine the length of the next stride so the robot approaches its target as fast as possible, yet without walking too far. Such a controller would be a crucial addition, since the robot could easily fall over if it failed to stop moving before picking up an object on a table, for example.

The robustness that can be added by making the walking pattern adjustable is essential for making the robot perform as naturally as possible in real-world situations. More research will be needed before the control systems can be implemented, however, to thoroughly evaluate the input-output relationships on such a high level.

Methods

Forward Kinematics Model of Left Leg

This section gives the forward kinematics model I developed for the robot's left leg. The right leg is almost identical, except that the offset from the waist is in the opposite direction. The angles and positions used below are defined as follows:

$\theta_{Hip,x}$ = sagittal angle of the hip,

$\theta_{Hip,y}$ = lateral angle of the hip,

$\theta_{Hip,z}$ = rotational angle of the hip,
$\theta_{Knee,x}$ = sagittal angle of the hip,
$\theta_{Ankle,x}$ = sagittal angle of the ankle,
$\theta_{Ankle,x}$ = lateral angle of the ankle,
$Foot_x$ = position of the bottom of the foot (directly beneath the ankle) in the x-direction,
$Foot_y$ = position of the bottom of the foot (directly beneath the ankle) in the y-direction,
$Foot_z$ = position of the bottom of the foot (directly beneath the ankle) in the z-direction.

The orientation angle of the Foot is not directly determined with this model. The coordinate system used to define the foot position is defined in figure 3. The origin is exactly halfway between the two hips. The positive x-direction points in the direction of walking, the positive y-direction points to the robot's left, and the positive z-direction points up.

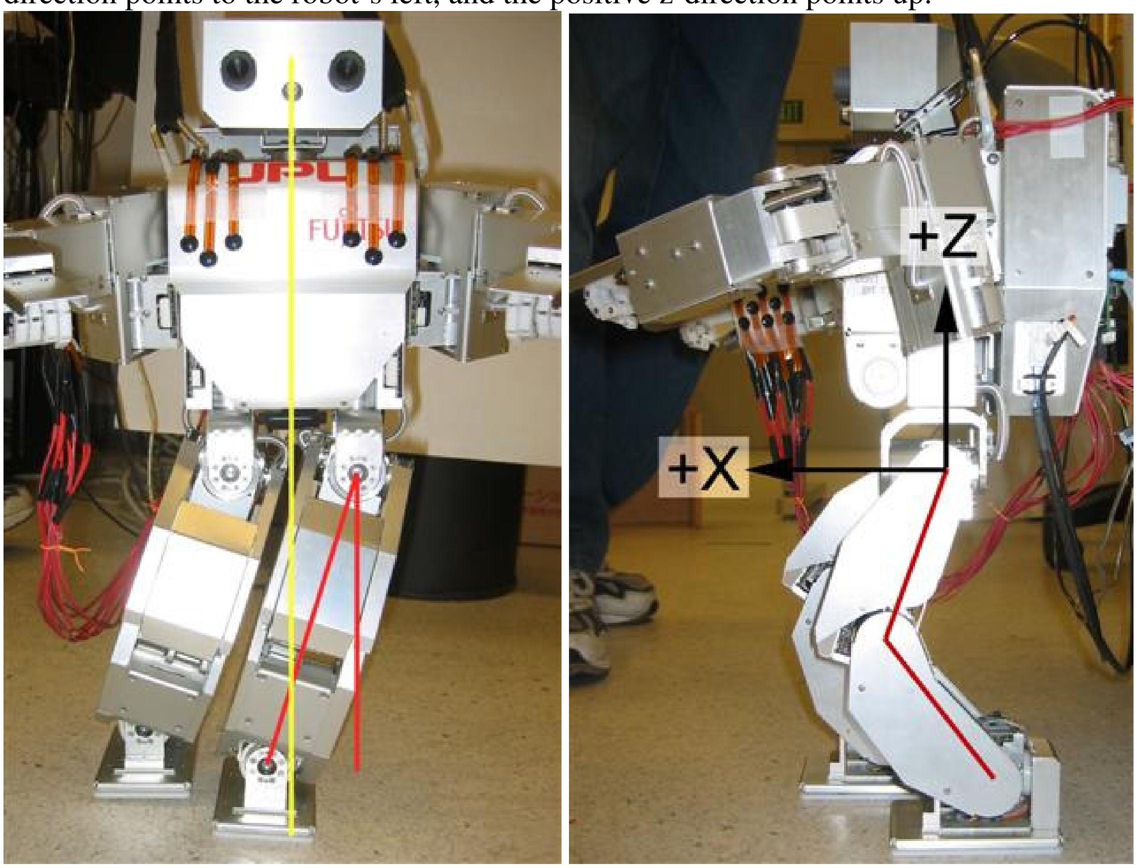

Figure 3: The coordinate system used to define foot positions. The origin is exactly halfway between the two hips. The positive x-direction points in the direction of walking, the positive y-direction points to the robot's left, and the positive z-direction points up.

The forward-kinematics matrix model is given in figure 4. This model is slightly simplified from that described in the Findings section, in that the foot orientation is not directly calculated, and the rotational motion of the hip is not considered, since that degree of freedom was not needed for the development of a basic walking algorithm.

$$\begin{bmatrix} Foot_x \\ Foot_y \\ Foot_z \\ Dummy \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(-\theta_{hip,y}) & -\sin(-\theta_{hip,y}) & 3.9 \\ 0 & \sin(-\theta_{hip,y}) & \cos(-\theta_{hip,y}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(-\theta_{hip,x}) & 0 & \sin(-\theta_{hip,x}) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(-\theta_{hip,x}) & 0 & \cos(-\theta_{hip,x}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot$$

$$\begin{bmatrix} \cos(-\theta_{knee,x}) & 0 & \sin(-\theta_{knee,x}) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(-\theta_{knee,x}) & 0 & \cos(-\theta_{knee,x}) & -10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos(-\theta_{ankle,x}) & 0 & \sin(-\theta_{ankle,x}) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin(-\theta_{ankle,x}) & 0 & \cos(-\theta_{ankle,x}) & -10 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(-\theta_{ankle,y}) & -\sin(-\theta_{ankle,y}) & 0 \\ 0 & \sin(-\theta_{ankle,y}) & \cos(-\theta_{ankle,y}) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 0 \\ 0 \\ -3.7 \\ 1 \end{bmatrix}$$

Figure 4: Matrix kinematics model for the robot's left leg. The fourth dimension is a "dummy" dimension which allows for the translation and transformation matrices to be combined into one matrix. The final vector defines the point on the foot (with respect to the ankle) that is converted to the central coordinate system. The orientation of the foot can be determined by changing the values of the final position vector.

Closed-Form Solution to Inverse-Kinematics Problem

Given the coordinates of the point on bottom of the foot directly below the ankle joint (with respect to the system defined in figure 1), and the restriction that the foot must be horizontal at all times, the solution to the inverse-kinematics problem in closed-form becomes much easier to determine. This section presents the derivation of that solution.

First, since the location hip joint is fixed with respect to the origin of the coordinate system, the coordinates can be translated to a system with its origin at the hip joint, 3.9cm away from the center of the waist,

$$Foot_{y,new} = Foot_y - 3.9cm\,.$$

Similarly, the foot being horizontal means that the link connecting the foot to the ankle joint must be vertical so the z-coordinate can be adjusted to represent the position of the ankle joint (3.7cm above the bottom of the foot), rather than the foot bottom,

$$Foot_{z,new} = Foot_z + 3.7cm\,.$$

The length of both the upper and lower legs is 10cm, thus the geometry of the leg can be considered to be the two congruent sides of an isosceles triangle and the third side to be the distance from the hip joint to the ankle joint,

$$Dist_{xyz} = \sqrt{Foot_x{}^2 + Foot_{y,new}{}^2 + Foot_{z,new}{}^2}\,.$$

The required angle of the knee joint (with respect to straight) to achieve this distance can now be easily calculated,

$$\theta_{knee,x} = \pi - 2\sin^{-1}\left(\frac{Dist_{xyz}}{2(10)}\right),$$

Consequently, the sagittal angle of the hip ($\theta_{hip,x}$) to satisfy this distance will have to be offset by

$$\theta_{hip,x,offset} = \frac{\theta_{knee,x}}{2}\,.$$

The total sagittal angle of the hip (to account for the x-position of the foot) can now be calculated,

$$\theta_{hip,x} = \tan^{-1}\left(\frac{Foot_x}{Foot_z}\right) + \theta_{hip,x,offset}.$$

Next, the lateral angle of the hip is calculated,

$$\theta_{hip,y} = \tan^{-1}\left(\frac{Foot_{y,new}}{Foot_{z,new}}\right).$$

Finally, using the flat foot restrictions, the remaining three angles are found,

$$\begin{aligned} \theta_{hip,z} &= 0 \\ \theta_{ankle,x} &= \theta_{hip,x} \\ \theta_{ankle,y} &= \theta_{knee,y} - \theta_{hip,y} \end{aligned}.$$

Acknowledgments

I would like to thank my mentor, Dr. Adrian Stoica and my co-mentor, Dr. Didier Keymeulen for sponsoring and supporting my research during my SURF at JPL. I would also like to thank my colleagues, Jiajing Xu and Tim Hidalgo for their work during the summer on this project.

References

[1] Lim, H., Yamamoto, Y., & Takanishi, A. Control to Realize Human-like Walking of a Biped Humanoid Robot. *IEEE International Conference on Systems, Man, and Cybernetics,* **5,** 3271-3276 (2000).
[2] Ogino, M., Katoh, Y., Asada, M., & Hosoda, K. Vision-Based Reinforcement Learning for Humanoid Behavior Generation with Rhythmic Walking Parameters. *Proceedings of 2003 IEEE/RSJ International Conference on Intelligent Robots and Systems,* **2,** 1665-71 (2003).
[3] Ogino, M., Hosoda, K., & Asada, M. Learning Energy Efficient Walking with Ballistic Walking. *2nd International Symposium on Adaptive Motion of Animals and Machines, March 2003.* Expected for Publication, Sept 2004.
[4] Stoica, A. Robot Fostering Techniques for Sensory-Motor Development of Humanoid Robots. *Robotics and Autonomous Systems,* **37,** 127-143 (2001).
[5] Tseng, H. C. Robotics I, *ELEN 371 Course Note, Santa Clara University* (1994).
[6] Huang, Q., Yokoi, K., Kajita, S., Kaneko, K., Arai, H., Koyachi, N., Tanie, K. Planning Walking Patterns for a Biped Robot. *IEEE Transactions on Robotics and Automation*, **17,** 280-289 (2001).
[7] Kagami, S., Kitagawa, T., Nishiwaki, K., Sugihara, T., Inaba, M., Inoue, H. A Fast Dynamically Equilibrated Walking Trajectory Generation Method of Humanoid Robot. *Autonomous Robots*, **12,** 71-82 (2002).
[8] Fujitsu Automation Co., Ltd. *HOAP-2 Instruction Manual* (2002).